HYBRID CRYPTOSYSTEM MENGGUNAKAN *XOR CIPHER* DAN *MERKLE–HELLMAN KNAPSACK* UNTUK MENJAGA KERAHASIAAN PESAN DIGITAL

Oris Krianto Sulaiman

Fakultas Teknik, Program Studi Teknik Informatika, Universitas Islam Sumatera Utara
Jl. Sisingamangaraja Teladan Barat, Medan, Indonesia
oris.ks@ft.uisu.ac.id

Abstract* - *The acceleration of data communication in the world of information technology is developing very fast, more than millions of digital messages are exchanging every day in cyberspace. These digital messages are freely communicating on internet networks or public networks that result in anyone being able to access these digital messages. Because the openness of the internet network that involves many people can take digital messages, the protection for the messages should be good. Digital message protection is used so that digital messages are directed to their intended destination, for this reason, a method is needed to maintain the confidentiality of the digital message. Cryptogafi is the study of encoding. By using cryptography digital messages can be changed into passwords so that the confidentiality of digital messages is maintained. Many algorithms can be used to change the original message (plaintext) into a password (ciphertext), one of which is the XOR cipher that utilizes the binary of each character. XOR cipher has a security hole that can be used by cryptanalysts such as using known-plaintext attack, ciphertext-only attack, and flip bit attack. Therefore, to increase the confidentiality of the message, the binary plaintext generated by the XOR cipher will be encrypted again using the Merkle-Hellman Knapsack. The results of this hybrid cryptosystem add a high level of digital message security because the binary ciphertext will re-form the ciphertext with Merkle-Hellman Knapsack, thus forming twice the protection.
Keywords* - Digital Message, *Hybrid Cryptosystem*, *XOR Cipher* dan *Merkle–Hellman Knapsack

Abstract* – Percepatan komunikasi data dalam dunia teknologi informasi berkembang sangat cepat, lebih dari jutaan pesan-pesan digital setiap harinya saling bertukar didunia maya. Pertukan pesan digital ini secara bebas berkomunikasi di jejaring internet atau jaringan umum yang mengakibatkan setiap orang dapat mengakses pesan digital tersebut. Sebab keterbukaan jaringan internet yang melibatkan banyak orang bisa mengambil pesan digital tersebut maka sudah seharusnya perlindungan untuk pesan tersebut harus baik. Perlindungan pesan digital tersebut digunakan agar pesan digital tertuju ke tujuan yang semestinya, untuk itu perlu adanya metode untuk menjaga kerahasiaan pesan digital tersebut. Kriptogafi merupakan ilmu yang mempelajari tentang penyandian. Dengan menggunakan kriptografi pesan digital dapat diubah menjadi sandi sehingga kerahasiaan pesan digital terjaga. Ada banyak algoritma yang dapat digunakan untuk merubah pesan asli *(plaintext)* menjadi sandi *(ciphertext)* salah satunya *XOR cipher* yang memanfaatkan *binary* dari setiap karakter. *XOR cipher* mempunyai celah keamanan yang dapat dimanfaatkan kriptanalis seperti menggunakan *known-plaintext attack*, *ciphertext-only attack* dan *flip bit attack*. Oleh sebab itu untuk meningkatkan kerahasiaan pesan maka *binary plaintext* yang dihasilkan oleh *XOR cipher* akan enkripsi lagi menggunakan *Merkle–Hellman Knapsack*. Hasil dari *hybrid cryptosystem* ini menambahkan tinggat keamanan pesan digital karena pada *binary ciphertex* akan membentuk *ciphertext* kembali dengan *Merkle–Hellman Knapsack*, sehingga membentuk dua kali perlindungan.
Kata kunci* – Pesan digital, *Hybrid Cryptosystem*, *XOR Cipher* dan *Merkle–Hellman Knapsack

I. PENDAHULUAN

Komunikasi data pada pesan digital saat ini merupakan hal yang sangat banyak ditemui disekitar. Kecepatan akses internet membantu proses komunikasi pesan digital tersebut berjalan dengan baik. Pesan-pesan digital ini saling berkomunikasi di internet, karena internet bersifat publik maka komunikasi pesan digital ini dapat dilihat oleh orang banyak. Hal ini menyebabkan ancaman bagi kerahasiaan pesan tersebut sehingga dibutuhkanlah teknik perlindungan untuk menjaga kerahasiaan tetap utuh [1], [2]. Teknik

kriptografi merupakan teknik yang dapat digunakna untuk menjaga kerahasiaan pesan agar tetap utuh. Teknik kriptografi memungkinkan pesan diubah kedalam sandi-sandi yang tidak dapat dibaca oleh orang lain. Dalam kriptografi pesan asli dikenal dengan sebutan *plaintext* dan pesan yang sudah menjadi sandi dikenal dengan sebutan *ciphertext.* Proses dari *plaintext* menjadi *ciphertext* disebut dengan enkripsi. Ada banyak algoritma enkripsi yang dapat digunakan untuk merubah *plaintext* menjadi *ciphertext* salah satunya adalah *XOR cipher.*

A. *XOR Cipher*

XOR cipher atau *Exlusive-OR cipher* merupakan teknik penyandian dengan memanfaatkan nilai *binary* dari setiap karakter pesan dan kunci. Masing-masing *binary plaintext* akan di XOR dengan *binary* kunci [3]– [5]. Adapun aturan dari dari operasi XOR dapat dilihat dari tabel 1 berikut:

Tabel 1. Operasi *XOR Cipher*

p	*k*	$c = p \oplus k$
0	0	0
0	1	1
1	0	1
1	1	0

p = Binary Plaintext;
k = Key;
c = Binary Plaintext.

untuk melakukan konversi karakter maka dibutuhkan *ASCII table* untuk melihat *binary* dari karakter A-Z.

Tabel 2. *Binary Code Converter*

Karakter	*ASCII Code*	*Binary*
A	065	0 1 0 0 0 0 0 1
B	066	0 1 0 0 0 0 1 0
C	067	0 1 0 0 0 0 1 1
D	068	0 1 0 0 0 1 0 0
E	069	0 1 0 0 0 1 0 1
F	070	0 1 0 0 0 1 1 0
G	071	0 1 0 0 0 1 1 1
H	072	0 1 0 0 1 0 0 0
I	073	0 1 0 0 1 0 0 1
J	074	0 1 0 0 1 0 0 1
K	075	0 1 0 0 1 0 1 1
L	076	0 1 0 0 1 1 0 0
M	077	0 1 0 0 1 1 0 1
N	078	0 1 0 0 1 1 1 0
O	079	0 1 0 0 1 1 1 1
P	080	0 1 0 1 0 0 0 0
Q	081	0 1 0 1 0 0 0 1
R	082	0 1 0 1 0 0 1 0
S	083	0 1 0 1 0 0 1 1
T	084	0 1 0 1 0 1 0 0
V	085	0 1 0 1 0 1 1 0
W	086	0 1 0 1 0 1 1 1
X	087	0 1 0 1 1 0 0 0
Y	088	0 1 0 1 1 0 0 1
Z	089	0 1 0 1 1 0 1 0

Dalam enkripsi *plaintext* menggunakan *XOR cipher* maka Panjang karakter *plaintext* harus sama dengan Panjang karakter kunci. Contoh penggunaan *XOR cipher* adalah sebagai berikut:
Plaintext: S R I E
Key: O R I S

Konversi *plaintext* dan *key* ke *binary* kemudian lakukan operasi XOR terhadap *binary plaintext* dan *binary key*. Proses enkripsi:
p = 01010011 01010010 01001001 01000101
k = 01001111 01010010 01001001 01010011
$\oplus$ ———————————————————
c = 00011100 00000000 00000000 00010110
Ciphertext = 29 0 0 22

Proses dekripsi:
c = 00011100 00000000 00000000 00010110
k = 01001111 01010010 01001001 01010011
$\oplus$ ———————————————————
p = 01010011 01010010 01001001 01000101
Plaintext = S R I E

B. *Merkle-Hellman Knapsack*

Merupakan algoritma kriptografi asimetris, dimana terdapat kunci *public* dan *private.* Mekanisme pengerjaan *merkle-hellman knapsack* adalah dengan menentukan *key generation* dari barisan *superincreasing.* Barisan *superincreasing* merupakan barisan di mana setiap nilai didalam barisan lebih besar daripada jumlah semua nilai sebelumnya [6]–[9].

$$w = (w_1, w_2, ..., w_n) \quad \text{.....} \quad (1)$$

w adalah barisan *superincreasing*. Berikutnya setiap elemen dalam barisan dikalikan dengan *r* modulo *q*

$$\beta_i = rw_i \bmod q. \quad \text{.....} \quad (2)$$

β merupakan kunci publik;
r merupakan bilangan yang bukan persekutuan dari *q*, atau bilangan prima;
q merupakan bilangan yang lebih besar daripada *w.*

Formula untuk melakukan enkripsi:

$$c = \sum_{i=1}^{n} \alpha_i \beta_i. \quad \text{.....} \quad (3)$$

Plaintext dikonversi kedalam *binary* yang pangjangnya sama dengan bit kuncil publik. Setiap bit *binary* dikalikan dengan yang berkorespondence di dalam kunci publik.
Untuk dekripsi maka dibutuhkan *c'* atau *invers*, untuk mencari *invers* dapat dilakukan dengan formula berikut:

$$c' = (1 + q*k)/r \quad \text{.....} \quad (4)$$

Formula untuk melakukan dekripsi:

$$c' = \sum_{i=1}^{n} \alpha_i w_i \quad \text{s .....} \quad (5)$$

Contoh dari penggunaan *merkle-hellman knapsack* adalah sebagai berikut [7]:
Plaintext: S R I E

w = (2, 3, 6, 13, 27, 52, 105, 210). *q* = 420, dan *r* = 23

Pembangkit kunci *public-private* :
2 * 23 mod 420 = 46 mod 420 = 46
3 * 23 mod 420 = 69 mod 420 = 69
6 * 23 mod 420 = 138 mod 420 = 138
13 * 23 mod 420 = 299 mod 420 = 299
27 * 23 mod 420 = 621 mod 420 = 201
Kunci Publik: (46, 69, 138, 299, 201, 356, 215,210).
Kunci *Private*: (2, 3, 6, 13, 27, 52, 105, 210)

Enkripsi:
Plaintext: S R I E
Binary: 01010011 01010010 01001001 01000101

Blok *plaintext*-1 = 01010011
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = (1*69)+ (1*299)+ (1*315)+ (1*210)
= 893

Blok *plaintext* -2 = 01010010
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = (1*69)+ (1*299)+ (1*315)
= 683

Blok *plaintext* -3 = 01001001
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = (1*69)+ (1*201)+ (1*210)
= 480

Blok *plaintext* -4 = 01000101
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = (1*69)+ (1*356)+ (1*210)
= 644

Ciphertext: 893 683 480 635

Dekripsi:
Ciphertext: 893 683 480 635

c' = (1+420**k*)/23
= (1+420*19)/23
= 347 (bilangan bulat)
Kunci *Private*: (2, 3, 6, 13, 27, 52, 105, 210)

Blok ciphertex-1 = 893
= 893 * 347 mod 420 = 331 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 01010011
= S

Blok ciphertex-2 = 683
= 683 * 347 mod 420 = 121 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 01010010
= R

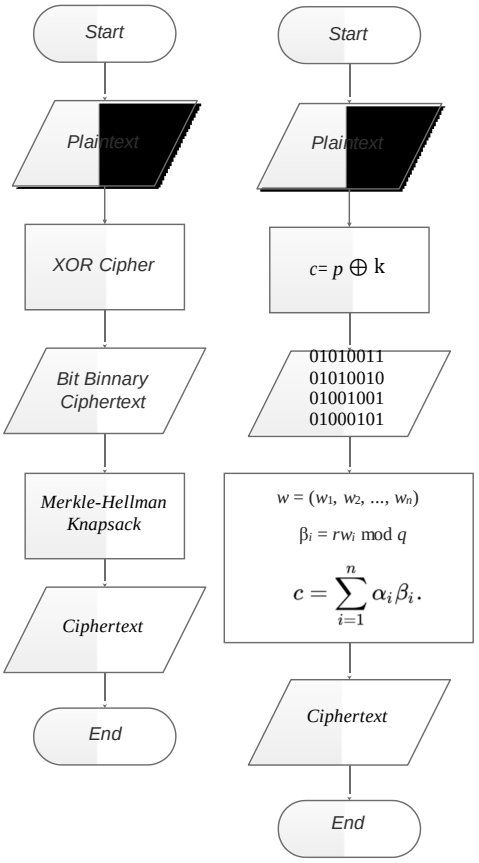
Blok ciphertex-3 = 480
= 480 * 347 mod 420 = 240 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 01001001
= I

Blok ciphertex-4 = 635
= 635 * 347 mod 420 = 265 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 01000101
= E
Plaintext = S R I E

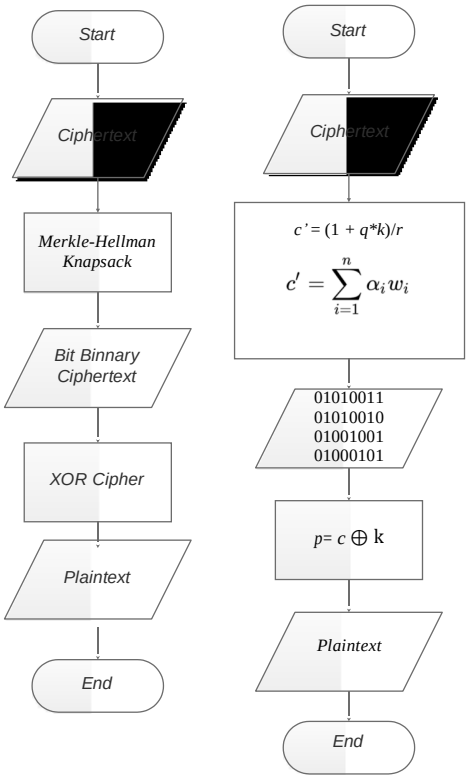
Adanya serangan serangan terhadap *XOR cipher* seperti menggunakan *known-plaintext attack, ciphertext-only attack* dan *flip bit attack* [10] dapat diatasi dengan menggunakan *merkle-hellman knapsack.*

II. METODE PENELITIAN

Untuk mengatasi permasalahan adanya celah keamanan dari *XOR cipher* yang dapat di serang menggunakan *known-plaintext attack, ciphertext-only attack* dan *flip bit attack*, maka metode penelitian dilakukan dengan menambil bit *binary ciphertext XOR Cipher* yang kemudian akan di enkripsi menggunakan *merkle-hellman knapsack* berdasarkan blok-blok bit *binary ciphertext XOR Cipher* sehingga hasil dari proses *Hybrid Cryptosystem* ini akan menghasilkan *ciphertext* berbentuk angka.
Untuk dekripsi merupakan inversi dari *ciphertext* yang kemudian *bit binary* di proses menggunakan *XOR Cipher* agar kembali ke *plaintext.*
Adapun *flowchart* metode penelitian ini dapat dilihat pada gambar berikut:



Gambar 1. Proses enkripsi *hybrid cryptosystem*



Gambar 2. Proses dekripsi *hybrid cryptosystem*

III. HASIL DAN PEMBAHASAN

Percobaan enkripsi *hybrid cryptosystem* dilakukan dengan menggunakan text sebagai berikut:
Plaintext: S R I E
Key: O R I S
Konversi *plaintext* dan *key* ke *binary* kemudian lakukan operasi XOR terhadap *binary plaintext* dan *binary key*.
Proses enkripsi:
p = 01010011 01010010 01001001 01000101
k = 01001111 01010010 01001001 01010011
⊕ ———————————————————
c = 00011100 00000000 00000000 00010110

w = (2, 3, 6, 13, 27, 52, 105, 210). *q* = 420, dan *r* = 23
Pembangkit kunci *public-private* :
2 * 23 mod 420 = 46 mod 420 = 46
3 * 23 mod 420 = 69 mod 420 = 69
6 * 23 mod 420 = 138 mod 420 = 138
13 * 23 mod 420 = 299 mod 420 = 299
27 * 23 mod 420 = 621 mod 420 = 201
Kunci Publik: (46, 69, 138, 299, 201, 356, 215,210).
Kunci *Private*: (2, 3, 6, 13, 27, 52, 105, 210)

Blok *plaintext*-1 = 00011100
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = (1*299)+ (1*201)+ (1*356)
= 856

Blok *plaintext* -2 = 00000000
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = 0

Blok *plaintext* -3 = 00000000
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = 0

Blok *plaintext* -4 = 00010110
Kunci publik= 46, 69, 138, 299, 201, 356, 315,210
Kriptogram = (1*299)+ (1*356)+ (1*315)
= 970

Ciphertext: 856 0 0 970

Untuk melakukan dekripsi
Ciphertext: 856 0 0 970

c' = (1+420**k*)/23
= (1+420*19)/23
= 347 (bilangan bulat)
Kunci *Private*: (2, 3, 6, 13, 27, 52, 105, 210)

Blok ciphertex-1 = 856
= 856 * 347 mod 420 = 92 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 00011100

Blok ciphertex-2 = 0
= 0 * 347 mod 420 = 0 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 0000000

Blok ciphertex-3 = 0
= 0 * 347 mod 420 = 0 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 0000000

Blok ciphertex-4 = 970
= 970 * 347 mod 420 = 170 mod 420
Koresponden = 2, 3, 6, 13, 27, 52, 105, 210
= 00010110

Bit Binary: 00011100 0000000 0000000 00010110

c = 00011100 00000000 00000000 00010110
k = 01001111 01010010 01001001 01010011
⊕ ———————————————————
p = 01010011 01010010 01001001 01000101
Plaintext = S R I E

Dari hasil percobaan yang telah dilakukan, *plaintext* berhasil di enkripsi menggunakan *XOR Cipher* yang kemudian menghasilkan *bit binary* yang digunakan oleh *merkle-hellman knapsack* sebagai *plaintext* yang kemudian di enkripsi kembali sehibgga menghasilkan angka *Ciphertext:* 856 0 0 970. *Chipertext* yang dihasilkan berupa angka yang dapat menutup kelemahan dari *XOR Cipher* berupa *bit binary.*

IV. KESIMPULAN

XOR cipher yang dapat di serang menggunakan *known-plaintext attack, ciphertext-only attack* dan *flip bit attack* dapat ditingkatkan kemanan pada *bit binary* dengan melakukan enkripsi pada *bit binary* menggunakan *merkle-hellman knapsack* sehingga *ciphertext* akan berubah menjadi angka. Sehingga serangan terhadap *bit binary XOR Cipher* lebih aman terhadap serangan.

DAFTAR PUSTAKA

[1] I. Febriana and G. A. S, "Penerapan Teknik Kriptografi Pada Keamanan Smsandroid," *JOEICT (Jurnal Educ. Inf. Commun. Technol.*, vol. 1, no. 1, pp. 29–36, 2017.

[2] M. M. Amin, "Implementasi Kriptografi Klasik Pada Komunikasi Berbasis Teks," *Pseudocode*, vol. 3, no. 2, pp. 129–136, 2017.

[3] D. S. Kumar, "A Block Cipher using Rotation and Logical XOR Operations," *Int. J. Comput. Sci. Issues*, vol. 8, no. 6, pp. 142–147, 2011.

[4] Suhardi, "Aplikasi Kriptografi Data Sederhana Dengan Metode Exlusive-or (Xor)," *Teknivasi*, vol. 03, pp. 23–31, 2016.

[5] A. P. Sidik *et al.*, "TEKNIK XOR PADA MODE OPERASI ALGORITMA CIPHER BLOCK CHAINING (CBC) DENGAN KUNCI ACAK BLUM BLUM SHUB DALAM MENINGKATKAN," vol. 3, no. 2, pp. 130–135, 2019.

[6] A. Aminudin, A. F. Helmi, and S. Arifianto, "Analisa Kombinasi Algoritma Merkle-Hellman Knapscak dan Logaritma Diskrit pada Aplikasi Chat," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 3, p. 325, 2018.

[7] A. Afni and O. K. Sulaiman, "KEAMANAN PESAN DENGAN MENGGUNAKAN KUNCI PUBLIK KNAPSACK CRYPTOSYSTEM 8 BIT," 2018.

[8] W. Zhang, B. Wang, and Y. Hu, "A new knapsack public-key cryptosystem," *5th Int. Conf. Inf. Assur. Secur. IAS 2009*, vol. 2, pp. 53–56, 2009.

[9] S. J. Aboud, "An improved knapsack public key cryptography system," *Int. J. Internet Technol. Secur. Trans.*, vol. 3, no. 3, pp. 310–319, 2011.

[10] R. Verdult, "Introduction to Cryptanalysis: Attacking Stream Ciphers," pp. 1–22, 2001.